SuperLU Implementation and Timings

CRS (with implementation help from Xiaoye Li), 8/6/03

- SuperLU is a sparse direct solver package that is part of the ACTS collection—source can be downloaded.
    - Three varieties: serial (Seq_SLU), distributed memory parallel (SLU_DIST), and shared memory.
    - Has minimum degree reordering but can use external ordering routines.
    - Does not assume symmetry.
    - Suggested by Dinesh Kaushik of the PETSc group.
- NIMROD implementation can link to either Seq_SLU or SLU_DIST.
    - Iter_cg_* modules now have routines to convert rblock matrix format to compressed column format (or LAPACK banded format).
    - Can be done for triangles.
    - New makefile system allows user to select the desired solver library.
    - "Link" libraries in the "externals" directory have C routines that interface SuperLU.

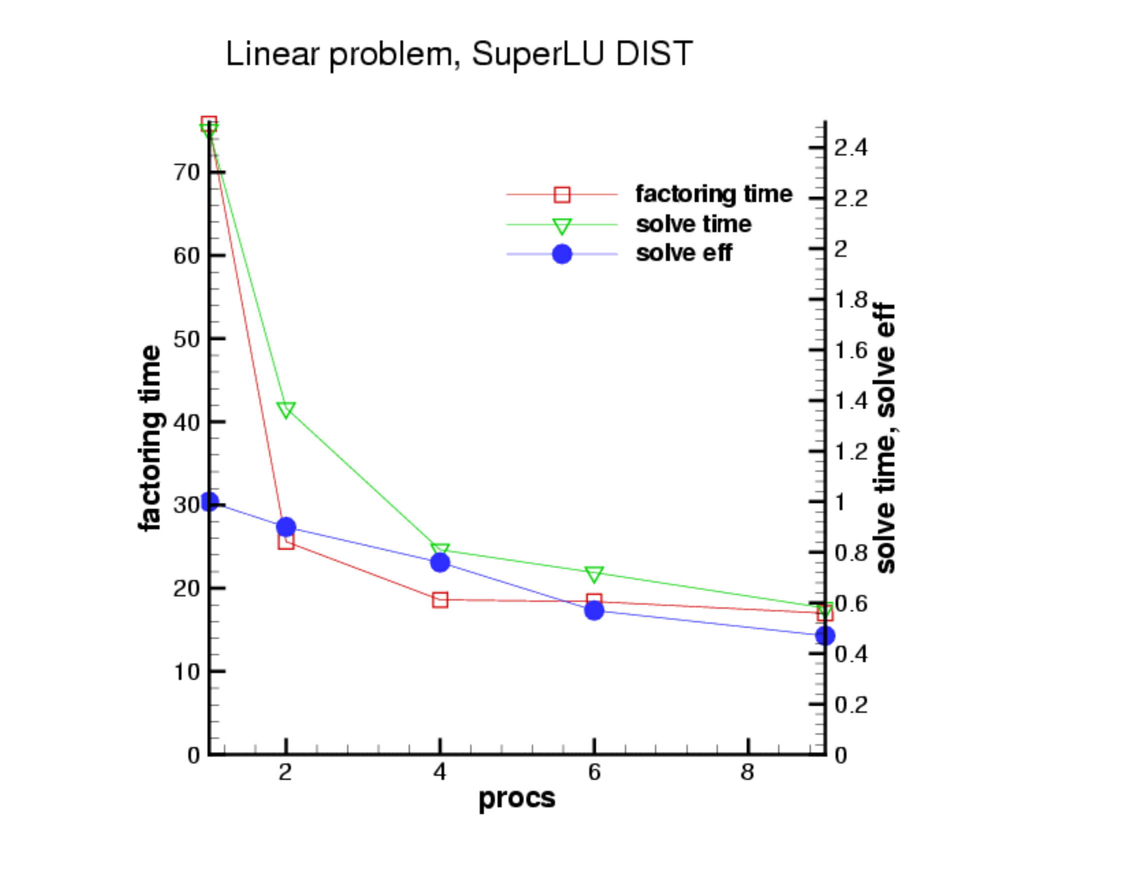
Timing Information

- With sufficient memory, serial linear computations can be 100 times faster than with our preconditioned CG.
  - Systems are ill-conditioned (direct is better but some iteration may still be required to achieve accuracy).
  - Our systems are small due to high-order elements.
  - A bicubic 32×32 linear computation runs 500 time-steps in about 12 minutes on our Pentium IV desktop.

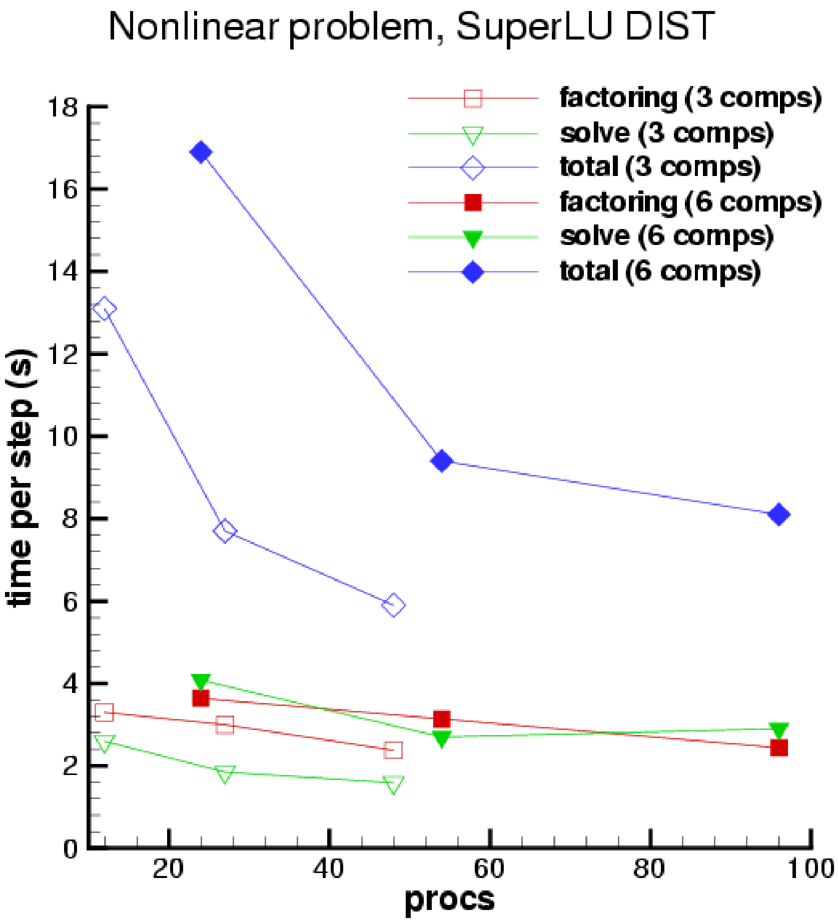
Parallel Timing Information

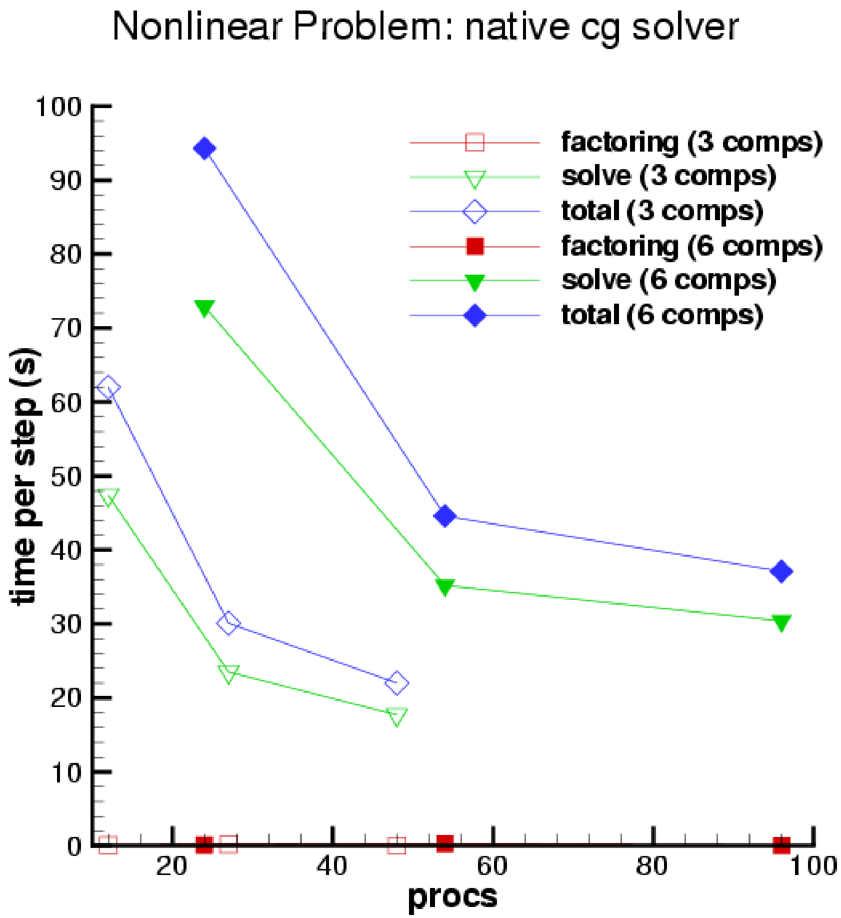
- Fixed-problem size parallel scaling is not as good as PCG, but the starting point is so much better that SLU seldom loses.

- 32×32 biquartic.
- Run on seaborg.



SLU_D is also effective for nonlinear problems.





- 32×32 biquartic D3D computation, all factors recomputed ~1 in 8 steps.
- Solve time includes 3D solves with SLU_D as the preconditioner.

Next Steps

- A fortran utility for direct nonsymmetric solves with SuperLU.
- Try a sparse solver designed for Hermitian systems.
    - But not SPOOLES—at least not yet (Esmond Ng).